Hardening Authentication *(Revised)*

On many networks, in order for users to be granted access to network resources, they must prove that they are who they say they are. This is the process of **authentication** of a user. The user can be authenticated by what he has (e.g., an ID card or token), what he knows (e.g., a password or PIN), or what he is (e.g., biometric data). More robust authentication processes use two or more of these factors.

Your network likely has an authentication mechanism in place to make sure that network resources cannot be accessed by unauthorized users. However, your authentication process may not be as robust as it should be, or your authentication mechanism itself may be vulnerable to attack. The following are some suggestions for hardening this critical piece of your infrastructure.

1. Limit Remote Access

Limiting remote access to your network is likely the most effective mitigation step you can take, because it will reduce your attack surface. Remote access should only be allowed for those users who truly need it to perform their duties; it should not be standard for all users. Of course, any systems used for remote access must be properly secured!

- Do not allow remote access clients to connect directly to the internal network – they should connect to a DMZ (demilitarized zone) and their traffic should be monitored.
- Restrict remote access to only authorized clients – for example, by filtering by IP address. Only company-owned systems with approved baselines should be allowed remote access.
- Limit concurrent logins to one per user.
- Audit login activity – verify suspicious logins with users, look for successful logins from unusual IP addresses, and look for spikes in failed logins.
- On each login, notify users of their last login date/time – if a user sees a suspicious last login, he should inform your network security personnel.

2. Augment Authentication Measures

If you are only using one factor (such as just a password) to authenticate your users, consider using a multifactor mechanism (such as PKI using a hardware token), especially for access to sensitive or critical resources and applications.

- Avoid transmitting authentication information using cleartext/weak protocols (e.g., telnet, ftp, http, pop3, ssh-1, etc.). Instead use only secure protocols as recommended by NIST FIPS or Special Publications (Available at http://csrc.nist.gov/publications/).
- Configure hosts to minimize the number of locally cached credentials. (Number of cached credentials should be set to 0 for desktops and servers, and 1 for laptops.)
- Set login restrictions by time and user type (e.g., no employee logins outside of normal work hours).
- For users who intermittently log in remotely, consider blocking their access by default – require them to first call in to be allowed access. Access should then automatically be blocked again after a defined amount of time.
- Implement a Network Access Protection/Control (NAP/NAC) solution to check the characteristics of a client machine before allowing it access to your network resources.
- Consider segmenting your network to isolate high-value assets and services, sensitive information, and privileged processes. If an intruder gains access, this will limit the damage that he can do.

3. Educate Users

Alert your users that they might receive phishing e-mails that ask them for their username, PIN, password, etc., or that direct them to unknown or untrusted web sites. To prevent their credentials from being stolen, users should never send any sensitive information in an e-mail, and they should immediately inform your network security personnel if they ever receive such requests.

If hardware credentials such as ID cards are used for authentication, remind your users to physically remove these credentials when not in use, to prevent an intruder from masquerading as an authenticated user.

Your users should never use high-privilege administrator accounts to browse the Internet or read e-mail. A malicious website, e-mail, or e-mail attachment could hijack the user's credentials, allowing an intruder to masquerade as the high-privilege user and do severe damage to the network. For suggestions on how to enforce this, see the "Enforcing No Internet or E-mail from Privileged Accounts" NSA Fact Sheet (Available at www.nsa.gov/ia/mitigation_guidance/security_configuration_guides/fact_sheets.shtml).

In general, authentication credentials that provide access to sensitive resources should never be used on any system that is also used to access the Internet – for example, the domain admin account should never be used to log onto and locally administer workstations.

The Information Assurance Mission at NSA

September 2012

4. Harden Authentication Server

Your authentication server should be hardened to prevent compromise of your authentication mechanism.

- The authentication server should be used solely for authentication; it should not also be used as a file server, web server, or anything else.
- Only install software verified as valid on the server. Software can be verified by checking its hash against the vendor-provided value, or by using digital signatures. If digital signatures are used, be sure to enable checking for revoked certificates using Online Certificate Status Protocol (OCSP) or Certificate Revocation List (CRL) checking.
- Document and periodically recheck the baseline install on the server, either manually or with a system integrity checking application.
- Change the default passwords on the server and disable unnecessary accounts.
- Prohibit Internet access to/from the server. Handle patches and updates through an offline process.
- Be sure the database of users that can be authenticated is up-to-date – old user accounts should be removed so they cannot be improperly used.
- Be sure all credentials are properly managed. Passwords should be stored securely – both on the server and on the clients – and transmitted securely (e.g., do not use Windows LanMan or any type of reversible encryption). Public key credentials should be validated to a known and trusted root and should be checked for revocation. Certificate trust stores and certificate usage should be minimized to those actually needed. Seeds for one-time password mechanisms should be protected with layered defenses.
    - For more information on password management, see NIST Special Publication 800-118: "Guide to Enterprise Password Management" (Available at http://csrc.nist.gov/publications/PubsSPs.html).
- Be sure all information about your authentication infrastructure (e.g., spreadsheets that link users to authentication data or that correlate different pieces of authentication data) is stored securely – preferably offline, or at least encrypted.
- If you are using a third party authentication solution, put that server in a separate security domain isolated from the rest of your network (e.g., for Windows Active Directory, the third party authentication server should not be part of the same forest as the rest of the network). An intruder on your network could use a temporary vulnerability to obtain domain credentials; isolating your authentication server reduces the risk that this intruder will then be able to obtain authentication information that would give him more persistent access to your network.
- Enable logging of all administrative actions done on your authentication server. Review these logs often, looking for any suspicious actions – especially any suspicious accesses of data used to authenticate users.
- Set firewall rules to restrict network and user access to the authentication server as much as possible. Only allow administrative access to the server from defined IP addresses within protected enclaves.
- Consider requiring physical access in order to administer the authentication server. Restrict physical access to only authorized administrators.

5. Establish Robust Authentication Policy

Although establishing a draconian password policy may be the easiest thing to do, it is also the least likely to be effective. Such a policy will only annoy your users and inspire them to develop ingenious ways to get around it. It is important, however, to have a robust yet reasonable policy.

- Consider requiring public key based authentication using FIPS certified hardware tokens.
- For password- and PIN-based accounts, enforce selection of robust passwords and PINs.
- Consider modeling user activity and setting thresholds to block anomalous login attempts. At a minimum, lock out a user after a reasonable number of failed login attempts. Consider a policy that requires your network security personnel to conduct a review before restoring access to a blocked account. A more sophisticated lock out policy should set multiple, incremental thresholds for failed authentication attempts, to distinguish potential Denial of Service attacks from inadvertent activity.
- Enforce the use of separate credentials for different accounts, especially for administrator accounts. If a user has administrative privileges, his administrative authentication credentials must be user-specific and different from his non-administrative authentication credentials.
- In general, the access allowed by a given authentication credential should be minimized, both in terms of which assets can be accessed and for how long they can be accessed (the latter can be accomplished by using a time-based access control mechanism such as Kerberos).